

**COMPUTATIONALLY EFFICIENT MEANS FOR OPTIMAL
CONTROL WITH CONTROL CONSTRAINTS**

- [1] This application claims priority to U.S. Provisional Application Serial No. 60/271,792, Filed February 27, 2001.

BACKGROUND OF THE INVENTION

- [2] Field of the Invention

This invention relates to optimal control of a system. More particularly, this invention relates to active vibration and active sound control systems for the interior of helicopters.

- [3] Background Art

Conventional active control systems consist of a number of sensors that measure the ambient variables of interest (*e.g.* sound or vibration), a number of actuators capable of generating an effect on these variables (*e.g.* by producing sound or vibration), and a computer which processes the information received from the sensors and sends commands to the actuators so as to reduce the amplitude of the sensor signals. The control algorithm is the scheme by which the decisions are made as to what commands to the actuators are appropriate.

- [4] A problem may arise in such a control scheme when the control decision yields a command that exceeds the physical capabilities of the system, for example, if the command to an actuator exceeds the actuator's physical limits.

SUMMARY OF THE INVENTION

- [5] The present invention utilizes a varying control weighting to provide improved performance while avoiding saturation of the actuators. For a least-squares or quadratic performance index, one typically includes a weighting on the control amplitudes to avoid actuator saturation. However, a constant weighting that is sufficient to ensure that no actuator ever saturates results in less than optimal performance. This invention therefore modifies the control weighting to ensure that the maximum possible performance is always obtained subject to the saturation constraints.

- [6] In the present invention, the ambient vibration is measured by a plurality of sensors. A plurality of command signals are generated over time based upon the measured vibration and based upon a control weighting. By varying the control weighting over time, the maximum possible performance is always obtained subject to the saturation constraints. The control weighting may be varied based upon the magnitude of the command signals.

BRIEF DESCRIPTION OF THE FIGURES

- [7] FIG. 1 shows a block diagram of the noise control system of the present invention.
[8] FIG. 2 shows a vehicle in which the present invention may be used.

DETAILED DESCRIPTION

- [9] Control systems consist of a number of sensors which measure ambient vibration (or sound), actuators capable of generating vibration at the sensor locations, and a computer which processes information received from the sensors and sends commands to the actuators which generate a vibration field to cancel ambient vibration (generated, for example by a disturbing force at the helicopter rotor). The control algorithm is the scheme by which the decisions are made as to what the appropriate commands to the actuators are.

- [10] FIG. 1 shows a block diagram 10 of an active control system. The system comprises a structure 102, the response of which is to be controlled, sensors 128, filter 112, control unit 106 and actuators (which could be force generators) 104. A disturbance source 103 produces undesired response of the structure 102. In a helicopter, for example, the undesired disturbances are typically due to vibratory aerodynamic loading of rotor blades, gear clash, or other source of vibrational noise. A plurality of sensors 128(a)...(n) (where n is any suitable number) measure the ambient variables of interest (e.g. sound or vibration). The sensors (generally 128) are typically microphones or accelerometers, or virtually any suitable sensors. Sensors 128 generate an electrical signal that corresponds to sensed sound or vibration. The electrical signals are transmitted to filter 112 via an associated interconnector 144(a)...(n) (generally 144). Interconnector 144 is typically wires or wireless transmission means, as known to those skilled in the art.

[11] Filter 112 receives the sensed vibration signals from sensors 128 and performs filtering on the signals, eliminating information that is not relevant to vibration or sound control. The output from the filter 112 is transmitted to control unit 106 via interconnector 142 respectively. The control circuit 106 generates control signals that control force generators 104(a)...(n).

[12] A plurality of force generators 104(a)...(n) (where n is any suitable number) are used to generate a force capable of affecting the sensed variables (e.g. by producing sound or vibration). Force generators 104(a)...(n) (generally 104) are typically speakers, shakers, or virtually any suitable actuators. Force actuators 104 receive commands from the control unit 106 via interconnector 134 and output a force, as shown by lines 132(a)...(n) to compensate for the sensed vibration or sound produced by vibration or sound source 103.

[13] The control unit 106 is typically a processing module, such as a microprocessor, with processing capabilities. Control unit 106 stores control algorithms control memory 105, or other suitable memory location. Memory module 105 is, for example, RAM, ROM, DVD, CD, a hard drive, or other electronic, optical, magnetic, or any other computer readable medium onto which is stored the control algorithms described herein. The control algorithms are the scheme by which the decisions are made as to what commands to the actuators 104 are appropriate.

[14] For a least-squares or quadratic performance index, one typically includes a weighting on the control amplitudes to avoid actuator saturation. However, a constant weighting that is sufficient to ensure that no actuator ever saturates results in less than optimal performance. This invention therefore modifies the control weighting to ensure that the maximum possible performance is always obtained subject to the saturation constraints. This is useful in the reduction of gear-mesh noise within a helicopter cabin.

[15] For tonal control problems, the computation can be performed at an update rate lower than the sensor sampling rate as described in the copending application entitled "Computationally Efficient Means for Active Control of Tonal Sound or Vibration," which is commonly assigned. This approach involves demodulating the sensor signals so that the desired information is near DC (zero frequency), performing the control computation, and remodulating the control commands to obtain the desired output to the actuators. The control computations are therefore performed on the sine and cosine components at the frequency of interest for each sensor signal.

These can be represented as a complex variable where the real part is equal to the cosine term, and the imaginary part is equal to the sine term. The same control algorithm can therefore be used for either low frequency disturbances or tonal disturbances, provided that complex notation is used. The approach will be described for such a demodulated tonal problem, but is equally applicable to low frequency disturbances.

[16] The number of sensors is given by n_s and the number of actuators n_a . The complex harmonic estimator variables that are calculated from the measurements of noise or vibration level can be assembled into a vector of length n_s denoted z_k at each sample time k . The control commands generated by the control algorithm can likewise be assembled into a vector of length n_a denoted u_k . The commands sent to the actuators are generated by multiplying the real and imaginary parts of this vector by the cosine and sine of the desired frequency.

[17] In the narrow bandwidth required for control about each tone, the transfer function between actuators and sensors is roughly constant, and thus, the system can be modeled as a single quasi-steady complex transfer function matrix, denoted T . This matrix of dimension n_s by n_a describes the relationship between a change in control command and the resulting change in the harmonic estimate of the sensor measurements, that is, $\Delta z_k = T \Delta u_k$. For notational simplicity, define $y_k = \Delta z_k$, and $v_k = \Delta u_k$. The complex values of the elements of T are determined by the physical characteristics of the system (including actuator dynamics, the structure and/or acoustic cavity, and anti-aliasing and reconstruction filters) so that T_{ij} is the response at the reference frequency of sensor i due to a unit command at the reference frequency on actuator j . Many algorithms may be used for making control decisions based on this model.

[18] The control law is derived to minimize a quadratic performance index

$$J = z^H W_z z + u^H W_u u + v^H W_{\delta u} v$$

where W_z , W_u and $W_{\delta u}$ are weighting matrices that are typically diagonal on the sensors, control inputs, and rate of change of control inputs respectively. A larger control weighting on an actuator will result in a control solution with smaller amplitude for that actuator.

[19] Solving for the control which minimizes J yields:

$$u_{k+1} = u_k - Y_k (W_u u_k + T_k^H W_z z_k)$$

where

$$Y_k = (T_k^H W_z T_k + W_u + W_{\delta u})^{-1}$$

[20] Solving for the steady state control ($u_{k+1} = u_k$) yields

$$u = -(T^H W_z T + W_u)^{-1} T^H W_z z_0$$

[21] The matrix Y determines the rate of convergence of different directions in the control space, but does not affect the steady state solution. In the following equation, the step size multiplier $\beta < 1$ provides control over the convergence rate of the algorithm. A value of approximately $\beta = 0.1$ may be used, for example.

$$u_{k+1} = u_k - \beta Y_k (W_u u_k + T_k^H W_z z_k)$$

[22] This equation corresponds to a recursive least-squares (RLS) control law. A least mean square (LMS) gradient approach could also be used, leading to a similar equation for u_{k+1} but with $Y = I$ instead. For poorly conditioned T matrices, the equalization of convergence rates for different directions that is obtained with the RLS approach is critical. Decreasing the control weighting, W_u , increases the low frequency gain, and decreasing the weighting on the rate of change of control, $W_{\delta u}$, increases the loop cross-over frequency (where frequency refers to the demodulated frequency). The notation $(u_i)_k$ will be used to refer to the (demodulated) command to the i^{th} actuator at the k^{th} control iteration.

[23] The command to any actuator may not exceed some (typically) fixed value U_{\max} , depending on the range of the D/A (digital to analog converter), amplifier limits, or actuation limits. This maximum value may be different for different actuators. If the desired command to an actuator $(u_i)_k$ as determined by the equations above exceeds the maximum allowable for that actuator by the ratio $R = |(u_i)_k| / U_{\max}$, then typically either that one particular actuator command is scaled by $1/R$, or the entire vector is scaled by $1/R$. In many applications the latter yields better results. For example, with two force actuators that combine to produce a force and a moment, it is preferable to reduce both actuator commands rather than just the saturated actuator, so that a significant change in the desired moment is not introduced. However, either of these two approaches can lead to a significant loss in performance.

[24] Typically, the control weighting values are fixed to a predefined value (based on experience), and are sometimes scheduled versus operating condition. However, in some applications such as using the active noise/vibration control algorithm to minimize helicopter

noise and/or vibration, it is very difficult to predetermine control weighting values that provide optimal performance over the entire flight envelope. Fixing the control weighting to prevent actuator saturation in the worst case condition will, in general, result in sub-optimal performance over the remainder of the flight envelope. Furthermore, the value of control weighting which results in optimal performance on one day may not yield the same performance on a different day, or even later during the same flight (*e.g.*, due to weight, temperature, or humidity changes). Based on the above discussion, it would be desirable to be able to actively vary the control weighting values to attempt to find the optimal values for the current operating condition.

[25] One aspect of the present invention is to introduce a time-varying control weighting. Another aspect is to improve the re-scaling of the actuator commands in the event of actuator saturation.

[26] Variable weighting

The approach used to achieve a time-varying control weighting in the noise and vibration control algorithm, denoted as variable control weighting, is as follows. When the code is initialized, the control weightings are set to some initial conservative values. As the control feedback loop converges to a steady state solution, the algorithm slowly varies the control weighting on a channel-by-channel basis until optimal performance is achieved. As discussed herein, "optimal performance" is defined as achieving the greatest reduction in the magnitude of the sensor signals z without saturating any of the actuator channels. In the variable control weighting approach, values of the actuator commands are used to actively adjust the control weighting values to maximize the reductions in the sensor signals while avoiding saturating any of the actuators. In a steady-state condition, the variable control weighting approach actively seeks for each actuator channel the smallest value of control weighting required to prevent saturation. However, recognizing the fact that a helicopter in flight is never truly in a steady-state condition, the variable control weighting approach actively increases the value of the control weighting for any actuator channel that saturates.

[27] The approach used in the noise and vibration control algorithm is to introduce various threshold levels, and multiply the control weighting $(W_{u,i})_k$ for each actuator i at each time step k by a scalar based on the magnitude of the control command $(u_i)_k$ computed for that actuator. A

table look-up was used for the reduction to practice in helicopter gear-mesh control, however, a continuous function could also have been used. An example is given by the table below.

| Magnitude of command | Scale control weight by |
|--|---|
| $0 < (u_i)_k < .5 * u_{\max}$ | $(W_{u,i})_{k+1} = 0.999 * (W_{u,i})_k$ |
| $.5 * u_{\max} < (u_i)_k < .8 * u_{\max}$ | $(W_{u,i})_{k+1} = 1.0 * (W_{u,i})_k$ |
| $.8 * u_{\max} < (u_i)_k < .9 * u_{\max}$ | $(W_{u,i})_{k+1} = 1.08 * (W_{u,i})_k$ |
| $.9 * u_{\max} < (u_i)_k < 1.0 * u_{\max}$ | $(W_{u,i})_{k+1} = 1.20 * (W_{u,i})_k$ |
| $1.0 * u_{\max} < (u_i)_k $ | $(W_{u,i})_{k+1} = 1.56 * (W_{u,i})_k$ |

The actual numbers used would differ for each specific application and would depend upon many different factors, including the update rate and the control bandwidth. Similarly, in some applications, the performance metric of interest is not the rms of all sensors, but rather the worst case. For example, with vibration control on a helicopter, the objective is to keep the vibration levels at all locations below some specified level. The weighting matrix W_z is used to place differential weighting on various sensors in which it is desired to produce reductions. In the case where there are specific maximum allowable sensor signal magnitudes to satisfy some predefined specification, such as a maximum vibration level in the case of AVC, the relative values of the elements of W_z can be actively adjusted such that all sensors are below their maximum allowable values. The relative values in the (typically diagonal) W_z matrix are adjusted such that the trace of the matrix remains unchanged but the element corresponding to the sensor which is above the desired specification is increased. This is done iteratively until all are below the desired specification.

Actuator scaling:

Conventional approaches to re-scaling the actuator command to avoid saturation result in less than optimal performance. The ideal approach is to modify the control weighting for the actuators (one or more) with the unallowable commands so that the new commands for those actuators are exactly at the limits. The invention described here accomplishes this ideal.

The nominal updated control step is given by

$$u_{k+1} = u_k - \beta(T_k^H W_z T_k + W_u + W_{\delta u})^{-1} (W_u u_k + T_k^H W_z z_k)$$

[28] If the i^{th} actuator command computed from this equation exceeds the allowable command u_{\max} (which may be different for different actuators), then a modified control step μ_{k+1} that satisfies $|\mu(i)_{k+1}| = u_{\max}$ can be computed in a computationally efficient manner from μ_{k+1} by substituting a new control weighting given by $W_u + \alpha e_i e_i^T$ into the control update equation. The notation $u(i)_k$ indicates the i^{th} element of the vector u at time step k . The vector e_i denotes a vector with all zero entries except for unity in the i^{th} entry, so that the new control weighting matrix differs from the original matrix only in the i^{th} diagonal entry. If there are multiple actuators at saturation ($L > 1$), then additional terms are added to W_u .

[29] Solving for μ_{k+1} for each actuator channel j yields

$$\mu(j)_{k+1} = u(j)_{k+1} - Y(j,i)\alpha(\mu(i)_{k+1} - (1-\beta)u(i)_k)$$

[30] The scalar $Y(j,i)$ refers to the $(j,i)^{\text{th}}$ element of the matrix Y defined earlier. For $L > 1$, the same equation holds with α being a real diagonal matrix of additional control weights, i representing a vector of indices corresponding to the saturated actuators, and $Y(j,i)$ a $1 \times L$ vector. From this equation, then given α , $\mu(i)_{k+1}$ can be computed. From this, the remaining elements of μ_{k+1} can be computed. The additional L control weightings on the i^{th} channels, α , and the L complex variables $\mu(j)_{k+1}$ are the solution of the L complex equations above along with the L real constraint equations that $|\mu(i)_{k+1}| = u_{\max}$. This can be solved in a variety of methods.

[31] In the case of a single unallowable command, α is the solution to the following scalar quadratic equation where $y = Y(i,i)$:

$$a(\alpha y)^2 + 2b(\alpha y) + c = 0$$

where

$$a = (1-\beta)^2 |u(i)_k|^2 - u_{\max}^2$$

$$b = (1-\beta)[u(i)_{k+1}^* u(i)_k + u(i)_k^* u(i)_{k+1}]/2 - u_{\max}^2$$

$$c = |u(i)_{k+1}|^2 - u_{\max}^2$$

[32] Note $a < 0$, since the command at k^{th} step can't exceed saturation and $c > 0$, since by presumption the command exceeds saturation. Therefore, regardless of the sign of b , there is a unique, real, positive solution α corresponding to the control weight for which the actuator with the nominal command above saturation would instead be exactly at saturation. If the new actuator command vector includes any actuators that exceed saturation, the approach can be

applied again in an iterative fashion. However, there is no guarantee that this process will converge. Therefore, since most of the benefit will accrue from the first or second use of this approach, the best compromise is to return to a simpler scaling of the command vector at some point if there are remaining actuator commands exceeding saturation. As an aside, also note that repeating the process for a second actuator would require re-computing the matrix Y with the new control weighting from the first iteration. A similar approach to that described here could be derived where several actuators were considered simultaneously; this would result in a set of coupled equations for the new control weightings.

[33] The scaling formulation described herein can be used in conjunction with the previously described variable control weighting approach. The final line of the table given in the variable control weighting section is modified to use the updated control weighting based on α . This approach results in optimal performance in the presence of actuator saturations. Without this additional modification, the variable control weighting described earlier will eventually converge to the optimal performance solution, however, the convergence may be poor. Note that if none of the actuators saturate in any conditions, then the actuators have more authority than they need. Hence, if the actuators must be designed so as to never saturate in order to ensure good performance, then the overall cost and weight of the system will be excessive.

[34] An alternate approach to scaling the control commands if one of them exceeds the saturation value is also possible. This alternate approach results in performance that is close to the optimal solution described above, although it constrains the control command for the saturated actuator more than is necessary. This approach involves scaling back the single command, computing the residual noise introduced by doing so, and computing an additional Δu for the remaining actuators to cancel this residual noise. This last step uses a large control rate weighting on the saturated actuator so that the additional Δu for that actuator is zero.

[35] With the first approach (suggested above), the appropriate control weight for that actuator is computed and can be used in the next step. Also, for large n_a , the computations required for the approach described below are substantially more than those required for the first approach. However, the second approach is guaranteed to converge if there are multiple actuator saturations. Therefore, the most appropriate algorithm for a particular application depends on the

number of actuators, the computation available, and the extent to which multiple actuators are frequently operating at their saturation limits. The second approach, described below, is described in more detail and claimed in copending U.S. Application Serial Number _____, filed _____, and also claims prior to U.S. Serial Number 60/271,792.

[36] Actuator Truncation Scheme

This approach to limiting the actuator commands guarantees convergence in n_a or less steps to a set of actuator commands which are at or below saturation. A nominal updated control step is given by:

$$u_{k+1} = u_k - \beta(T_k^H W_z T_k + W_u + W_{\delta u})^{-1}(W_u u_k + T_k^H W_z z_k)$$

[37] If the i^{th} actuator command computed from this equation exceeds the allowable command u_{\max} (which may be different for different actuators), then a modified control update μ_{k+1} that satisfies $|\mu(i)_{k+1}| = u_{\max}$ is computed from u_{k+1} by scaling $u(i)_{k+1}$ by the ratio $u_{\max}/|u(i)_{k+1}|$ while all other elements of u_{k+1} are held fixed. Then a revised control update u^*_{k+1} is computed from the following control update equation:

$$u^*_{k+1} = \mu_{k+1} - (T_k^H W_z T_k + W^*_{\delta u} + \gamma e_i e_i^T)^{-1}(T_k^H W_z T_k(\mu_{k+1} - u_{k+1}))$$

with

$$W^*_{\delta u} = W_{\delta u} + \gamma e_i e_i^T$$

where γ is "a large number" (*i.e.*, on the order of 10^6 times larger than the magnitude of a typical element of $W_{\delta u}$.) The vector e_i denotes a vector with all zero entries except for unity in the i^{th} entry, so that the new control rate weighting matrix differs from the original matrix only in the i^{th} diagonal entry. The addition of the term $\gamma e_i e_i^T$ to the i^{th} element of the control rate weighting matrix $W_{\delta u}$ prevents the i^{th} element of μ_{k+1} from being altered by the above control update equation, *i.e.* it is guaranteed that $u^*(i)_{k+1} = \mu(i)_{k+1}$

[38] The above procedure is repeated iteratively for each saturated channel (one channel at a time) until no elements of the control update exceed their respective saturation values. This iterative procedure is guaranteed to converge in no more than n_a steps since once an actuator channel has reached saturation, that channel's command is held fixed at saturation until the next

control update cycle. Once this procedure has been completed, then $u_{k+1} = u_{k+1}^*$ is defined as the control update for the current cycle.

[39] FIG. 2 shows a perspective view 20 of a vehicle 118 in which the present invention can be used. Vehicle 118, which is typically a helicopter, has rotor blades 119 (a)...(d). Gearbox housing 110 is mounted at an upper portion of vehicle 118. Gearbox mounting feet 140 (a)...(c) (generally 140) provide a mechanism for affixing gearbox housing 110 to vehicle airframe 142. Sensors 128(a) through (d) (generally 128) are used to sense acoustic vibration produced by the vehicle, which can be from the rotorblades 119 or the gearbox housing 110. Although only four sensors are shown, there are typically any suitable number of sensors necessary to provide sufficient feedback to the controller (not shown). The sensors 128 may be mounted in the vehicle cabin, on the gearbox mounting feet 140, or to the airframe 142, or to another location on the vehicle 118 that enables vehicle vibrations or acoustic noise to be sensed. Sensors 128 are typically microphones, accelerometers or other sensing devices that are capable of sensing vibration produced by gear clash from the gearbox 110 and generating a signal as a function of the sensed vibration. These sensors generate electrical signals (voltages) that are proportional to the local noise or vibration.

[40] In accordance with the provisions of the patent statutes and jurisprudence, exemplary configurations described above are considered to represent a preferred embodiment of the invention. However, it should be noted that the invention can be practiced otherwise than as specifically illustrated and described without departing from its spirit or scope. Note that alphanumeric labels on method steps are for clarifying references in dependent claims and unless otherwise specified do not require a specific sequence in which the steps are to be performed.